BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

Google™ Advanced Search Preferences Language Tools Search Tips

verilog (standard or specification) Google Search

The word **"or"** was ignored in your query -- for search results including one term or another, use capitalized "OR" between words. [details]

Web · Images · Groups · Directory · News ·

Searched the web for **verilog (standard or specification or syntax)**. Results **1** - **10** of about **5,660**. Search took

[PDF] Update on the new **Verilog**-1995 **Standard**
File Format: PDF/Adobe Acrobat - View as HTML
... to a formal **specification**) An evolved **standard** (not designed to a formal **specification**) »» Inconsistent **syntax** & ... some data in the **Verilog** HDL Cannot ...
www.sutherland-hdl.com/papers/ 1995-IVC-1364_PLI_standard_status.pdf - Similar pages

[PDF] Transitioning to the New PLI **Standard**
File Format: PDF/Adobe Acrobat - View as HTML
... products Supported by all major **Verilog** simulator products ... **Standard** The "Old" PLI **Standard** □ □ The TF ... Not developed to a **specification** Not developed to ...
www.sutherland-hdl.com/papers/ 1997-CUG-presentation_switching_to_vpi.pdf - Similar pages
[More results from www.sutherland-hdl.com]

Accelera
... SCE-API) 1.0, SystemVerilog 3.1 and **Verilog**-AMS 2.1 ... the shortcomings of natural language forms of **specification**. ... gives the design architect a **standard** means of ...
www.accellera.org/press19.html - 28k - Nov 20, 2003 - Cached - Similar pages

Accelera
... the unification of Accellera's Property **Specification** Language (PSL ... Accellera **Verilog** Analog/Mixed Signal (**Verilog**-AMS) design language **standard** with its ...
www.accellera.org/press21.html - 17k - Nov 20, 2003 - Cached - Similar pages
[More results from www.accellera.org]

[PDF] Synopsys, Inc Desirable features for **Standard** Accellera Property ...
File Format: PDF/Adobe Acrobat - View as HTML
... 1 Desirable features for **Standard** Accellera Property **Specification** Language Clocking ... Compatibility Boolean Expressions 1. **Verilog** compatible boolean ...
www.eda.org/vfv/hm/att-0675/01-SynopsysFeedback.pdf - Similar pages

HDL Planet's **Verilog** Page
... expected that the IEEE would ratify the **standard** in 2000 ... http://home.europa.com/~celiac/VerilogBNF.html **Verilog** Formal **Syntax Specification** obtained from ...
hdlplanet.tripod.com/verilog/verilog.html - 12k - Cached - Similar pages

[DOC] remarques sur PT54V
File Format: Microsoft Word 6 - View as HTML
... knowing the validation level reached by the **specification**. ... errors (error to be corrected, **standard** ambiguity, case ... V3.0.0 geodecheck (c) **VERILOG** 1988 1996 ...
docbox.etsi.org/MTS/MTS/03-Permdoc/96/mts96_10.doc - Similar pages

Verilog HDL
... **Verilog** simulation semantics (from IEEE 1364 Draft document: DRAFT **STANDARD VERILOG**

h g gec e ch h e e e g

HDL); **Verilog** Formal **Syntax Specification** (based on Language Reference Manual **...**
www.cl.cam.ac.uk/users/mjcg/Verilog/ - 10k - Cached - Similar pages

VERILOG LINKS

... **Verilog** simulation semantics from IEEE 1364 Draft document: DRAFT **STANDARD VERILOG** HDL); **Verilog** Formal **Syntax Specification** (based on Language Reference Manual **...**
www.deeps.org/verilog/verilinks.html - 31k - Cached - Similar pages

[PDF] Microsoft PowerPoint - 05_Accellera_update.ppt
File Format: PDF/Adobe Acrobat
... Semantics Charter: Develop a **standard syntax** and semantics **...** that addresses requirements **specification** for systems **...** SystemVerilog Charter: Extend **Verilog** IEEE 2001 **...**
www.synopsys.com/partners/tapin/forumpres/ oct2003/Accellera_update.pdf - Similar pages

Goooooooooogle ▶
Result Page: **1** 2 3 4 5 6 7 8 9 10 **Next**

verilog (standard or specification | Google Search | Search within results

Dissatisfied with your search results? Help us improve.

Google Home - Advertise with Us - Business Solutions - Services & Tools - Jobs, Press, & Help

h g gec e ch h e e e e g

Google Advanced Search Preferences Language Tools Search Tips

verilog force release Google Search

Web · Images · Groups · Directory · News ·

Searched the web for **verilog force release**. Results **1** - **10** of about **4,490**. Search took **0.10** seconds.

gEDA: Icarus **Verilog** 20000421 Snapshot
... on the -D and the -I flags, but I expect this program to replace the **verilog**.sh script before the 0.3 **release**. And *finally* I got started on **force/release**. ...
www.geda.seul.org/mailinglist/geda-dev29/msg00075.html - 6k - Cached - Similar pages

gEDA: Icarus **Verilog** Version 0.2 **Release** Candidate 1
... Most of these are actually very closely related within Icarus **Verilog**, so make ... 4) ``assign a = b;'' is a one way operation, 5) **force/release**/assign/deassign ...
www.geda.seul.org/mailinglist/geda-dev27/msg00072.html - 7k - Cached - Similar pages
[More results from www.geda.seul.org]

/bbs/pub/vhdlsynth/reflector: **Force/release**/get signals in hier
... I hate to say this, but **Verilog** has an ... string, value: std_logic_vector); procedure **force**(path : string, value: std_logic); procedure **release**(path : string ...
www.eda.org/vhdlsynth/reflector/hm/0092.html - 6k - Cached - Similar pages

/bbs/pub/vhdlsynth/reflector: Re: **Force/release**/get signals in
... Can't **force** errors or view internal signals without the **force/release**/get, such ... the flow is easy to understand in > fork and join.> Unlike **Verilog** where always ...
www.eda.org/vhdlsynth/reflector/hm/0102.html - 6k - Cached - Similar pages
[More results from www.eda.org]

gEDA: Icarus **Verilog** 20011103 Snapshot
... **verilog**-20011103.tar.gz> <ftp://icarus.com/pub/eda/**verilog**/snapshots/**verilog**-20011103.txt> I'm thinking that after I add support for **force/release**, I'll make ...
archives.seul.org/geda/dev/Nov-2001/msg00004.html - 7k - Cached - Similar pages

[DOC] CDA 6214 Structured VLSI Design
File Format: Microsoft Word 97 - View as HTML
... register or net. **Verilog** provides two procedural continuous assignment options: assign/deassign and **force/release**. assign/deassign. ...
www.cse.fau.edu/~abhi/ch9.doc - Similar pages

Sample Solutions to Exercise 1 Chapter 2 1.
... Modules can be nested to any arbitrary depth - subject to the machine on which **Verilog** is support. See Page 45 of the textbook. i. Yes, using **force**...**release**. ...
www.cs.nthu.edu.tw/~tcwang/4120-spring03/ exercise-sol-1.txt - 16k - Cached - Similar pages

Accelera
NEWS **RELEASE** ... the EDA organization resulting from the unification of Open **Verilog** International (OVI ... VI); today announced the formation of task **force** to define ...
www.accellera.org/press1.html - 16k - Nov 20, 2003 - Cached - Similar pages

Accelera
NEWS **RELEASE** ... Advanced Library Format, Design Constraints, consistent ASIC function and timing support via VITAL and **Verilog** HDL ASIC Task **Force**, and the ...
www.accellera.org/press4.html - 17k - Nov 20, 2003 - Cached - Similar pages
[More results from www.accellera.org]

h g gec e ch ce c e e e e g f c

Verilog
Summary of Supported **Verilog** Constructs: **...** random; $stop; $strobe[b,h,o]; $time; $write; $write[b,h,o]; $curphase; $cycles. **force**, **release** statements; **...**
www.quickturn.com/products/ssverilog.htm - 94k - Cached - Similar pages

Gooooooooogle ▶
Result Page: **1** 2 3 4 5 6 7 8 9 10 **Next**

verilog force release Google Search Search within results

Dissatisfied with your search results? Help us improve.

Google Home - Advertise with Us - Business Solutions - Services & Tools - Jobs, Press, & Help

h g g e c e ch ce c e e e e g f c

Welcome to IEEE Xplore®
- Home
- What Can I Access?
- Log-out

Tables of Contents
- Journals & Magazines
- Conference Proceedings
- Standards

Search
- By Author
- Basic
- Advanced

Member Services
- Join IEEE
- Establish IEEE Web Account
- Access the IEEE Member Digital Library

1) Enter a single keyword, phrase, or Boolean expression. Example: acoustic imaging (means the phrase acoustic imaging plus any stem variations)
2) Limit your search by using search operators and field codes, if desired.
Example: optical (fiber fibre) ti
3) Limit the results by selecting Search Options.
4) Click Search. See Search Examples

((module* or sub*routine* or header* or function* or procedure*) <paragraph> ((initial* or assign* or set* or defin* or pre*determin*) <paragraph> ((initial or starting or first or beginning) <sentence>

Start Search Clear

Note: This function returns plural and suffixed forms of the keyword(s).

Search operators: More

Field codes: au (author), ti (title), ab (abstract), jn (publication name), de (index term) More

Search Options:

Select publication types:
- [x] IEEE Journals
- [x] IEE Journals
- [x] IEEE Conference proceedings
- [x] IEE Conference proceedings
- [x] IEEE Standards

Select years to search:
From year: All to Present

Organize search results by:
Sort by: Relevance
In: Descending order
List 15 Results per page

Your search matched **242** of **987057** documents.

A maximum of **242** results are displayed, **15** to a page, sorted by **Relevance** in **descending** order.
You may refine your search by editing the current search expression or entering a new one the text box.

Then click **Search Again**.

((module* or sub*routine* or header* or function* or procedure*) <paragraph> ((initial*

Search Again

Results:
Journal or Magazine = **JNL** Conference = **CNF** Standard = **STD**

1 **Application of the distortion method of model validation using the PC-based simulation language, Berkeley Madonna**
Thomas, P.J.;
Model Validation for Plant Control and Condition Monitoring (Ref. No. 2000/044), IEE Seminar on , 28 March 2000
Page(s): 6/1 -6/8

[Abstract] [PDF Full-Text (472 KB)] **IEE CNF**

2 **Portable acquisition and identification tool for induction machine diagnosis**
Teodorescu, R.; Tataru, A.M.; Lungeanu, F.; Iov, F.; Dumitriu, T.;
Power Electronics and Variable Speed Drives, 1998. Seventh International Conference on (IEE Conf. Publ. No. 456) , 21-23 Sept. 1998
Page(s): 515 -520

[Abstract] [PDF Full-Text (328 KB)] **IEE CNF**

3 **Frequency tracking by adaptive notch filter structures**
Kwan, K.; Martin, J.D.;
New Directions in Adaptive Signal Processing, IEE Colloquium on , 16 Feb 1993
Page(s): 4/1 -4/4

[Abstract] [PDF Full-Text (172 KB)] **IEE CNF**

4 **Experimental investigations of floating-gate circuits for /spl Delta/ - /spl Sigma/ modulators**
Pereira, A.; Brady, P.; Bandyopadhyay, A.; Hasler, P.E.;
Circuits and Systems, 2002. MWSCAS-2002. The 2002 45th Midwest Symposium on , Volume: 1 , 4-7 Aug. 2002
Page(s): I -208-11 vol.1

[Abstract] [PDF Full-Text (316 KB)] **IEEE CNF**

---

5 **The design of time-varying digital filters which employ binary valued coefficients**
Kitson, F.; Griffiths, L.;
Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '82. , Volume: 7 , May 1982
Page(s): 302 -305

[Abstract] [PDF Full-Text (112 KB)] **IEEE CNF**

---

6 **Adaptive nonlinear image restoration by a modified Kalman filtering approach**
Rajala, S.; de Figueiredo, R.;
Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '80. , Volume: 5 , Apr 1980
Page(s): 414 -417

[Abstract] [PDF Full-Text (160 KB)] **IEEE CNF**

---

7 **High resolution model based 2-D spectrum estimation**
Hansen, R.R., Jr.; Chellappa, R.;
Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988 International Conference on , 11-14 April 1988
Page(s): 733 -736 vol.2

[Abstract] [PDF Full-Text (364 KB)] **IEEE CNF**

---

8 **On the application of hidden Markov models for enhancing noisy speech**
Ephraim, Y.; Malah, D.; Juang, B.-H.;
Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988 International Conference on , 11-14 April 1988
Page(s): 533 -536 vol.1

[Abstract] [PDF Full-Text (428 KB)] **IEEE CNF**

---

9 **Minimal power consumption of the Penn State rollerscrew**

EVAD through optimal and suboptimal control
Klute, G.K.; Tsach, U.; Geselowitz, D.B.;
Bioengineering Conference, 1988., Proceedings of the 1988 Fourteenth Annual Northeast , 10-11 March 1988
Page(s): 74 -78

[Abstract] [PDF Full-Text (292 KB)] **IEEE CNF**

---

10 **The incremental-cost approach for synthesis of CCD 4-valued unary functions**
Abd-El-Barr, M.H.; Hoang, T.D.; Vranesic, Z.G.;
Multiple-Valued Logic, 1988., Proceedings of the Eighteenth International Symposium on , 24-26 May 1988
Page(s): 82 -89

[Abstract] [PDF Full-Text (476 KB)] **IEEE CNF**

---

11 **Modular state variable simulation of spacecraft power systems including nonlinear inductances**
Reid, B.B.; Evans, B.W.; Nelms, R.M.; Grigsby, L.L.;
System Theory, 1989. Proceedings., Twenty-First Southeastern Symposium on , 26-28 March 1989
Page(s): 127 -131

[Abstract] [PDF Full-Text (284 KB)] **IEEE CNF**

---

12 **On identification of nonstationary Hammerstein systems by the Fourier series regression estimate**
Krzyzak, A.;
Decision and Control, 1989., Proceedings of the 28th IEEE Conference on , 13-15 Dec. 1989
Page(s): 626 -629 vol.1

[Abstract] [PDF Full-Text (264 KB)] **IEEE CNF**

---

13 **A new autosizing algorithm for CMOS combinational logic circuits**
Chung-Yu Wu; Jen-Sheng Hwang;
VLSI Technology, Systems and Applications, 1989. Proceedings of Technical Papers. 1989 International Symposium on , 17-19 May 1989
Page(s): 242 -246

[Abstract] [PDF Full-Text (444 KB)] **IEEE CNF**

---

14 **Fetal heart rate detection by a special transform method**
Tal, Y.; Akselrod, S.;
Computers in Cardiology 1989. Proceedings. , 19-22 Sept. 1989
Page(s): 275 -278

[Abstract] [PDF Full-Text (308 KB)] **IEEE CNF**

---

15 **Numerical integration methods based on variation diminishing polynomial approximation for the solution of algebraic differential equations in the context of circuit simulation**
Yanilmaz, M.;
Circuits and Systems, 1989., Proceedings of the 32nd Midwest Symposium on , 14-16 Aug. 1989
Page(s): 454 -457 vol.1

[Abstract] [PDF Full-Text (296 KB)] **IEEE CNF**

Your search matched **18** of **987057** documents.

A maximum of **18** results are displayed, **25** to a page, sorted by **Relevance** in **descending** order.
You may refine your search by editing the current search expression or entering a new one the text box.

Then click **Search Again**.

(vhdl or verilog or hdl) and ((module* or header*) <paragraph> ((force* or initial* or ass

Search Again

Results:
Journal or Magazine = **JNL** Conference = **CNF** Standard = **STD**

1 **Analog design with Verilog-A**
Miller, I.; FitzPatrick, D.; Aisola, R.;
Verilog HDL Conference, 1997., IEEE International , 31 March-2 April 1997
Page(s): 64 -68

[Abstract] [PDF Full-Text (328 KB)] **IEEE CNF**

2 **SDV/sup 2/: dynamic visualization of VERILOG simulations**
Marczynski, R.; Seidel, P.M.;
Circuits and Systems, 2002. MWSCAS-2002. The 2002 45th Midwest Symposium on , Volume: 1 , 4-7 Aug. 2002
Page(s): I -148-51 vol.1

[Abstract] [PDF Full-Text (360 KB)] **IEEE CNF**

3 **Fast prototyping of an ASIC for ATM application using a synthesizable VHDL flexible library**
Claretto, S.; Filippi, E.; Montanaro, A.; Paolini, M.; Turolla, M.;
VHDL International Users' Forum, 1997. Proceedings , 19-22 Oct. 1997
Page(s): 88 -94

[Abstract] [PDF Full-Text (588 KB)] **IEEE CNF**

4 **Digital design from concept to prototype in hours**
Hsu, Y.C.; Liu, T.Y.; Tsai, F.S.; Lin, S.Z.; Yu, C.;

Circuits and Systems, 1994. APCCAS '94., 1994 IEEE Asia-Pacific Conference on , 5-8 Dec. 1994
Page(s): 175 -181

[Abstract] [PDF Full-Text (648 KB)] **IEEE CNF**

---

5 **A purely behavioral data structure for accurate high level timing simulation of synchronous designs**
Arnold, M.G.; Bailey, T.A.; Cowles, J.R.; Cupal, J.J.; Wallace, A.W.;
Verilog HDL Conference, 1994., International , 14-16 March 1994
Page(s): 101 -107

[Abstract] [PDF Full-Text (448 KB)] **IEEE CNF**

---

6 **Getting started with VHDL**
Jones, P.L.;
Microelectronic Systems Education, 1997. MSE '97. Proceedings., 1997 IEEE International Conference on , 21-23 July 1997
Page(s): 135 -136

[Abstract] [PDF Full-Text (120 KB)] **IEEE CNF**

---

7 **A reusable microcontroller core's design**
Janiszewki, I.; Baraniecki, R.; Siekierska, K.;
Fall VIUF Workshop, 1999. , 4-6 Oct. 1999
Page(s): 14 -19

[Abstract] [PDF Full-Text (636 KB)] **IEEE CNF**

---

8 **A VHDL implementation of a shearing unit for shear-warp factorization volume rendering**
Kazakova, N.V.; Margala, M.;
Electrical and Computer Engineering, 2000 Canadian Conference on , Volume: 2 , 7-10 March 2000
Page(s): 1118 -1122 vol.2

[Abstract] [PDF Full-Text (312 KB)] **IEEE CNF**

---

9 **A portable mechanism for vectorizing compiled event-driven simulation**
Mittra, S.;
Verilog HDL Conference, 1997., IEEE International , 31 March-2 April 1997
Page(s): 70 -76

[Abstract] [PDF Full-Text (456 KB)] **IEEE CNF**

10 **Register-transfer level fault modeling and test evaluation techniques for VLSI circuits**
Thaker, P.A.; Agrawal, V.D.; Zaghloul, M.E.;
Test Conference, 2000. Proceedings. International , 3-5 Oct. 2000
Page(s): 940 -949

[Abstract] [PDF Full-Text (788 KB)] **IEEE CNF**

11 **A flexible architecture for H.263 video coding**
Garrido, M.J.; Sanz, C.; Jimenez, M.; Meneses, J.M.;
Digital System Design, 2002. Proceedings. Euromicro Symposium on , 4-6 Sept. 2002
Page(s): 70 -77

[Abstract] [PDF Full-Text (1595 KB)] **IEEE CNF**

12 **Teaching computer hardware design using commercial CAD tools**
Puvvada, G.; Breuer, M.A.;
Education, IEEE Transactions on , Volume: 36 Issue: 1 , Feb. 1993
Page(s): 158 -163

[Abstract] [PDF Full-Text (616 KB)] **IEEE JNL**

13 **A VLSI architecture for lifting-based forward and inverse wavelet transform**
Andra, K.; Chakrabarti, C.; Acharya, T.;
Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on] , Volume: 50 Issue: 4 , April 2002
Page(s): 966 -977

[Abstract] [PDF Full-Text (422 KB)] **IEEE JNL**

14 **A test evaluation technique for VLSI circuits using register-transfer level fault modeling**
Thaker, P.A.; Agrawal, V.D.; Zaghloul, M.E.;
Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on , Volume: 22 Issue: 8 , Aug. 2003
Page(s): 1104 -1113

[Abstract] [PDF Full-Text (525 KB)] **IEEE JNL**

15 **A system level HW/SW partitioning and optimization tool**
Schwiegershausen, M.; Kropp, H.; Pirsch, P.;
Design Automation Conference, 1996, with EURO-VHDL '96 and Exhibition, Proceedings EURO-DAC '96, European , 16-20 Sept. 1996
Page(s): 120 -125

[Abstract] [PDF Full-Text (796 KB)] **IEEE CNF**

16 **Using RASSP modules in a rapid system prototyping class**
Landis, D.L.;
Microelectronic Systems Education, 1999. MSE '99. IEEE International Conference on , 19-21 July 1999
Page(s): 53 -54

[Abstract] [PDF Full-Text (20 KB)] **IEEE CNF**

17 **Ethernet network-based DAQ and smart sensors for the OPERA long-baseline neutrino experiment**
Girerd, C.; Gardien, S.; Burch, J.; Katsanevas, S.; Marteau, J.;
Nuclear Science Symposium Conference Record, 2000 IEEE , Volume: 2 , 15-20 Oct. 2000
Page(s): 12/111 -12/115 vol.2

[Abstract] [PDF Full-Text (488 KB)] **IEEE CNF**

18 **Continuously live image processor for drift chamber track segment triggering**
Berenyi, A.; Chen, H.K.; Dao, K.; Dow, S.F.; Gehrig, S.K.; Gill, M.S.; Grace, C.; Jared, R.C.; Johnson, J.K.; Karcher, A.; Kasen, D.; Kirsten, F.A.; Kral, J.F.; LeClerc, C.M.; Levi, M.E.; von der Lippe, H.; Liu, T.H.; Marks, K.M.; Meyer, A.B.; Minor, R.;
Nuclear Science, IEEE Transactions on , Volume: 46 Issue: 3 , June 1999
Page(s): 348 -353

[Abstract] [PDF Full-Text (604 KB)] **IEEE JNL**

Your search matched **13** of **985183** documents.

A maximum of **13** results are displayed, **25** to a page, sorted by **Relevance** in **descending** order.
You may refine your search by editing the current search expression or entering a new one the text box.

Then click **Search Again**.

(vhdl or verilog or hdl) and ((module* or header*) <paragraph> (initialize or assign or set

Search Again

Results:
Journal or Magazine = **JNL** Conference = **CNF** Standard = **STD**

1 **Fast prototyping of an ASIC for ATM application using a synthesizable VHDL flexible library**
Claretto, S.; Filippi, E.; Montanaro, A.; Paolini, M.; Turolla, M.;
VHDL International Users' Forum, 1997. Proceedings , 19-22 Oct. 1997
Page(s): 88 -94

[Abstract] [PDF Full-Text (588 KB)] **IEEE CNF**

2 **Digital design from concept to prototype in hours**
Hsu, Y.C.; Liu, T.Y.; Tsai, F.S.; Lin, S.Z.; Yu, C.;
Circuits and Systems, 1994. APCCAS '94., 1994 IEEE Asia-Pacific Conference on , 5-8 Dec. 1994
Page(s): 175 -181

[Abstract] [PDF Full-Text (648 KB)] **IEEE CNF**

3 **A reusable microcontroller core's design**
Janiszewki, I.; Baraniecki, R.; Siekierska, K.;
Fall VIUF Workshop, 1999. , 4-6 Oct. 1999
Page(s): 14 -19

[Abstract] [PDF Full-Text (636 KB)] **IEEE CNF**

4 **A VHDL implementation of a shearing unit for shear-warp factorization volume rendering**
Kazakova, N.V.; Margala, M.;

Electrical and Computer Engineering, 2000 Canadian Conference on , Volume: 2 , 7-10 March 2000
Page(s): 1118 -1122 vol.2

[Abstract] [PDF Full-Text (312 KB)] **IEEE CNF**

---

5 **A portable mechanism for vectorizing compiled event-driven simulation**
Mittra, S.;
Verilog HDL Conference, 1997., IEEE International , 31 March-2 April 1997
Page(s): 70 -76

[Abstract] [PDF Full-Text (456 KB)] **IEEE CNF**

---

6 **Register-transfer level fault modeling and test evaluation techniques for VLSI circuits**
Thaker, P.A.; Agrawal, V.D.; Zaghloul, M.E.;
Test Conference, 2000. Proceedings. International , 3-5 Oct. 2000
Page(s): 940 -949

[Abstract] [PDF Full-Text (788 KB)] **IEEE CNF**

---

7 **A flexible architecture for H.263 video coding**
Garrido, M.J.; Sanz, C.; Jimenez, M.; Meneses, J.M.;
Digital System Design, 2002. Proceedings. Euromicro Symposium on , 4-6 Sept. 2002
Page(s): 70 -77

[Abstract] [PDF Full-Text (1595 KB)] **IEEE CNF**

---

8 **Teaching computer hardware design using commercial CAD tools**
Puvvada, G.; Breuer, M.A.;
Education, IEEE Transactions on , Volume: 36 Issue: 1 , Feb. 1993
Page(s): 158 -163

[Abstract] [PDF Full-Text (616 KB)] **IEEE JNL**

---

9 **A VLSI architecture for lifting-based forward and inverse wavelet transform**
Andra, K.; Chakrabarti, C.; Acharya, T.;
Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on] , Volume: 50 Issue: 4 ,

April 2002
Page(s): 966 -977

[Abstract] [PDF Full-Text (422 KB)] **IEEE JNL**

---

10 **A test evaluation technique for VLSI circuits using register-transfer level fault modeling**
Thaker, P.A.; Agrawal, V.D.; Zaghloul, M.E.;
Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on , Volume: 22 Issue: 8 , Aug. 2003
Page(s): 1104 -1113

[Abstract] [PDF Full-Text (525 KB)] **IEEE JNL**

---

11 **A system level HW/SW partitioning and optimization tool**
Schwiegershausen, M.; Kropp, H.; Pirsch, P.;
Design Automation Conference, 1996, with EURO-VHDL '96 and Exhibition, Proceedings EURO-DAC '96, European , 16-20 Sept. 1996
Page(s): 120 -125

[Abstract] [PDF Full-Text (796 KB)] **IEEE CNF**

---

12 **Using RASSP modules in a rapid system prototyping class**
Landis, D.L.;
Microelectronic Systems Education, 1999. MSE '99. IEEE International Conference on , 19-21 July 1999
Page(s): 53 -54

[Abstract] [PDF Full-Text (20 KB)] **IEEE CNF**

---

13 **Ethernet network-based DAQ and smart sensors for the OPERA long-baseline neutrino experiment**
Girerd, C.; Gardien, S.; Burch, J.; Katsanevas, S.; Marteau, J.;
Nuclear Science Symposium Conference Record, 2000 IEEE , Volume: 2 , 15-20 Oct. 2000
Page(s): 12/111 -12/115 vol.2

[Abstract] [PDF Full-Text (488 KB)] **IEEE CNF**

Your search matched **[0]** of **[987057]** documents.

You may refine your search by editing the current search expression or entering
a new one the text box. Then click search Again.

(vhdl or verilog or hdl) and ((module* or header*) <paragraph> (initialize) <paragraph>

Search Again

OR

Use your browser's back button to return to your original search page.

Results:

No documents matched your query.

Search Results

Search Results for: **[verilog <paragraph> initial* <paragraph> module*]**
Found **29** of **123,929 searched.**

Search within Results

> Advanced Search

> Search Help/Tips

Sort by: Title Publication Publication Date Score Binder

Results 1 - 20 of 29 short listing

Prev Page 1 2 Next Page

1 Xilinx FPGA design tools for Linux 100%
Michael Baxter
Linux Journal October 2003
Volume 2003 Issue 114
Walk through a hardware design cycle with new tools that are bringing electronic design automation to our favorite platform.

2 Architectural power estimation and optimization: Clock-tree power optimization based on RTL clock-gating 100%
Monica Donno , Alessandro Ivaldi , Luca Benini , Enrico Macii
Proceedings of the 40th conference on Design automation June 2003
As power consumption of the clock tree in modern VLSI designs tends to dominate, measures must be taken to keep it under control. This paper introduces an approach for reducing clock power based on clock gating. We present a methodology that, starting from an RTL description, automatically generates a set of constraints for driving the construction of the clock tree by the clock synthesis tool. The methodology has been fully integrated into an industry-strength design flow, based on Synopsys Des ...

3 Novel techniques in high-level synthesis: Automating the design of an asynchronous DLX microprocessor 100%
Manish Amde , Ivan Blunno , Christos P. Sotiriou
Proceedings of the 40th conference on Design automation June 2003
In this paper the automated design of an asynchronous DLX microprocessor is presented. The microprocessor has been designed beginning with a standard RTL-like *Verilog* specification and the *Pipefitter* design flow has been used to automatically generate both the specification for the direct implementation of the Control Unit and a synthesisable Verilog specification of the Data Path. The architecture of the DLX is locally synchronous and globally asynchronous

and the delay elements f ...

4 CAD: A comprehensive high-level synthesis system for control-flow intensive behaviors 100%

W. Wang , T. K. Tan , J. Luo , Y. Fei , L. Shang , K. S. Vallerio , L. Zhong , A. Raghunathan , N. K. Jha

Proceedings of the 13th ACM Great Lakes Symposium on VLSI April 2003

In this paper, we describe a comprehensive high-level synthesis system for control-flow intensive as well as data-dominated behaviors. We propose a new control-data flow graph model to preserve the parallelism inherent in the application, as well as to facilitate high-level synthesis. Our algorithm, which is based on an iterative improvement strategy, performs clock selection, scheduling, module selection, resource allocation and assignment simultaneously to fully derive the benefits of design s ...

5 Tutorial: Compiling concurrent languages for sequential processors 100%

Stephen A. Edwards

ACM Transactions on Design Automation of Electronic Systems (TODAES) April 2003

Volume 8 Issue 2

Embedded systems often include a traditional processor capable of executing sequential code, but both control and data-dominated tasks are often more naturally expressed using one of the many domain-specific concurrent specification languages. This article surveys a variety of techniques for translating these concurrent specifications into sequential code. The techniques address compiling a wide variety of languages, ranging from dataflow to Petri nets. Each uses a different method, to some degr ...

6 Icarus verilog: open-source verilog more than a year later 100%

Stephen Williams , Michael Baxter

Linux Journal July 2002

Volume 2002 Issue 99

More competitive than ever--and still free!

7 Formal verification: Handling special constructs in symbolic simulation 100%

Alfred Kölbi , James Kukula , Kurt Antreich , Robert Damiano

Proceedings of the 39th conference on Design automation June 2002

Symbolic simulation is a formal verification technique which combines the flexibility of conventional simulation with powerful symbolic methods. Some constructs, however, which are easy to handle in conventional simulation need special consideration in symbolic simulation. This paper discusses some special constructs that require unique treatment in symbolic simulation such as the symbolic representation of arrays, an efficient This paper discusses some special constructs that are unique to symb ...

8 Moving towards more effective validation: Deriving a simulation input generator and a coverage metric from a formal specification 100%

Kanna Shimizu , David L. Dill

Proceedings of the 39th conference on Design automation June 2002

This paper presents novel uses of functional interface specifications for verifying RTL designs. We demonstrate how a simulation environment, a correctness checker, and a functional coverage metric are all created automatically from a single specification. Additionally, the process exploits the structure of a specification written with simple style rules. The

methodology was used to verify a large-scale I/O design from the Stanford FLASH project.

9 High level specification and design: RTL c-based methodology for designing and verifying a multi-threaded processor 100%

Luc Sèmèria , Renu Mehra , Barry Pangrle , Arjuna Ekanayake , Andrew Seawright , Daniel Ng

Proceedings of the 39th conference on Design automation June 2002

A RTL C-Based design and verification methodology is presented with enabled the successful high speed validation of a 7 million gate simultaneous multi-threaded (SMT) network processor. The methodology is centered on statically scheduled C-based coding style, C to HDL translation, and a novel RTL-C to RTL-Verilog equivalence checking flow. It leverages improved simulation performance combined with static techniques to reduce the amount of RTL-Verilog and gate-level verification required during d ...

10 High level specification and design: Formal verification of module interfaces against real time specifications 100%

Arindam Chakrabarti , Pallab Dasgupta , P. P. Chakrabarti , Ansuman Banerjee

Proceedings of the 39th conference on Design automation June 2002

One of the main concerns of the designer of a circuit module is to guarantee that the interface of the module conforms to specific protocols (such as PCI Bus, AMBA bus or Ethernet) by which it interacts with its environment. The computational complexity of verifying such open systems under all possible environments has been shown to be very hard (EXPTIME complete [10]). On the other hand, designers are typically required to guarantee correct behavior only for specific valid behaviors of the envi ...

11 IP Design and Reuse: Concurrent-simulation-based remote IP evaluation over the internet for system-on-a-chip design 100%

Hung-Pin Wen , Chien-Yu Lin , Youn-Long Lin

Proceedings of the 14th international symposium on Systems synthesis September 2001

We propose an Internet-based concurrent-simulation scheme to ease IP evaluation process between IP vendors and users. Complex system-on-a-chip design requires more and more IP modules from 3rd party vendors. What can be disclosed by the vendor without impairing its trade secrete and what needs to be examined by the user to gain satisfactory level of confidence are contradictory of each other. Via PLI interface functions and Internet protocol, our proposed software enables HDL simulators (Verilog ...

12 Formal property verification by abstraction refinement with formal, simulation and hybrid engines 100%

Dong Wang , Pei-Hsin Jiang , James Kukula , Yunshan Zhu , Tony Ma , Robert Damiano

Proceedings of the 38th conference on Design automation June 2001

We present RFN, a formal property verification tool based on abstraction refinement. Abstraction refinement is a strategy for property verification. It iteratively refines an abstract model to better approximate the behavior of the original design in the hope that the abstract model alone will provide enough evidence to prove or disprove the property.However, previous work on abstraction refinement was only demonstrated on designs with up to 500 registers. We developed RFN to ver ...

13 Assertion checking by combined word-level ATPG and modular arithmetic constraint-solving 100%

techniques
Chung-Yang Huang , Kwang-Ting Cheng
Proceedings of the 37th conference on Design automation June 2000
We present a new approach to checking assertion properties for RTI, design verification. Our approach combines structural, word-level automatic test pattern generation (ATPG) and modular arithmetic constraint-solving techniques to solve the constraints imposed by the target assertion property. Our word-level ATPG and implication technique not only solves the constraints on the control logic, but also propagates the logic implications to the datapath. A novel arithmetic constraint so ...

14 Piranha: a scalable architecture based on single-chip multiprocessing 100%
Luiz André Barroso , Kourosh Gharachorloo , Robert McNamara , Andreas Nowatzyk , Shaz Qadeer , Barton Sano , Scott Smith , Robert Stets , Ben Verghese
ACM SIGARCH Computer Architecture News , Proceedings of the 27th annual international symposium on Computer architecture May 2000
Volume 28 Issue 2
The microprocessor industry is currently struggling with higher development costs and longer design times that arise from exceedingly complex processors that are pushing the limits of instruction-level parallelism. Meanwhile, such designs are especially ill suited for important commercial applications, such as on-line transaction processing (OLTP), which suffer from large memory stall times and exhibit little instruction-level parallelism. Given that commercial applications constitute by fa ...

15 Superlog, a unified design language for system-on-chip 100%
Peter L. Flake , Simon J. Davidmann
Proceedings of the 2000 conference on Asia South Pacific design automation January 2000

16 Verischemelog: Verilog embedded in Scheme 100%
James Jennings , Eric Beuscher
ACM SIGPLAN Notices , Proceedings of the 2nd conference on Domain-specific languages December 1999
Volume 35 Issue 1
Verischemelog (pronounced with 5 syllables, veruh-scheme-uh-log) is a language and programming environment embedded in Scheme for designing digital electronic hardware systems and for controlling the simulation of these circuits. Simulation is performed by a separate program, often a commercial product. Verischemelog compiles to Verilog, an industry standard language accepted by several commercial and public domain simulators.Because many ...

17 Simulation vector generation from HDL descriptions for observability-enhanced statement coverage 100%
Farzan Fallah , Pranav Ashar , Srinivas Devadas
Proceedings of the 36th ACM/IEEE conference on Design automation conference June 1999

18 Vex—A CAD toolbox 100%
Jules P. Bergmann , Mark A. Horowitz
Proceedings of the 36th ACM/IEEE conference on Design automation conference June

1999

19 A video signal processor for MIMD multiprocessing 100%
Jörg Hilgenstock , Klaus Herrmann , Jan Otterstedt , Dirk Niggemeyer , Peter Pirsch
Proceedings of the 35th annual conference on Design automation conference May 1998

20 ATM traffic shaper: ATS 100%
J. C. Diaz , P. Plaza , J. Crespo
Proceedings of the conference on Design, automation and test in Europe February 1998
The design and Implementation of an ATM Traffic Shaper (ATS) is here described. This IC was realized on a 0.35m CMOS technology. The main function of the ATS is the collection of low bit rate traffics to fill a higher bit rate pipe in order to reduce the cost of ATM based services, nowadays mainly influenced by transmission cost. The circuit fits in several ATM system configurations but mainly will be used at the User-Network Interfaces or Network-Network interfaces. The IC was designed with a T ...

Search Results

Search Results for: **[verilog <paragraph> initial* <paragraph> module*]**
Found **29** of **123,929 searched.**

Search within Results

> Advanced Search

> Search Help/Tips

Sort by: Title Publication Publication Date Score Binder

Results 21 - 29 of 29 short listing

Prev Page 1 2 Next Page

21 Toward formalizing a validation methodology using simulation coverage 100%
Aarti Gupta , Sharad Malik , Pranav Ashar
Proceedings of the 34th annual conference on Design automation conference June 1997

22 Preserving HDL synthesis hierarchy for cell placement 100%
Yu-Wen Tsay , Wen-Jong Fang , Allen C.-H. Wu , Youn-Long Lin
Proceedings of the 1997 international symposium on Physical design April 1997

23 An observability-based code coverage metric for functional simulation 100%
Srinivas Devadas , Abhijit Ghosh , Kurt Keutzer
Proceedings of the 1996 IEEE/ACM international conference on Computer-aided design January 1997
Functional simulation is the most widely used method for design verification. At various levels of abstraction, e.g., behavioral, register-transfer level and gate level, the designer simulates the design using a large number of vectors attempting to debug and verify the design. A major problem with functional simulation is the lack of good metrics and tools to evaluate the quality of a set of functional vectors. Metrics used currently are based on instruction counts and are quite simplistic. Des ...

24 A system for compiling and debugging structured data processing controllers 100%
A. Seawright , J. Buck , U. Holtmann , W. Meyer , B. Pangrle , R. Verbrugghe
Proceedings of the conference on European design automation September 1996

25 Verification of electronic systems 100%
Alberto L. Sangiovanni-Vincentelli , Patrick C. McGeer , Alexander Saldanha
Proceedings of the 33rd annual conference on Design automation conference June 1996

26 A general method for compiling event-driven simulations 100%
Robert S. French , Monica S. Lam , Jeremy R. Levitt , Kunle Olukotun
Proceedings of the 32nd ACM/IEEE conference on Design automation conference January 1995

27 Dynamic scheduling and synchronization synthesis of concurrent digital systems under system-level constraints 100%
Claudionor N. Coelho , Giovanni De Micheli
Proceedings of the 1994 IEEE/ACM international conference on Computer-aided design November 1994
We present in this paper a novel control synthesis technique for system-level specifications that are better described as a set of concurrent synchronous descriptions, their synchronizations and constraints. The proposed synthesis technique considers the degrees of freedom introduced by the concurrent models and by the environment in order to satisfy the design constraints.Synthesis is divided in two phases. In the first phase, the original specification is translated into an alg ...

28 Evaluation criteria of HDLs: VHDL compared to Verilog, UDL/I & M 100%
Serge Maginot
Proceedings of the conference on European design automation November 1992

29 An engineering environment for hardware/software co-simulation 100%
D. Becker , R. K. Singh , S. G. Tell
Proceedings of the 29th ACM/IEEE conference on Design automation conference July 1992

Search Results

Search Results for: **[((module* or sub*routine* or header* or function* or procedure*) <paragraph> ((initial* or assign* or set* or defin* or pre*determin*) <paragraph> ((initial or starting or first or beginning) <sentence> (node* or signal* or variable* or wire* or port* or reg* or register*) <sentence> (condition* or value* or voltage* or state*)))))]**
Found **2,417** of **123,929 searched.**

Warning: Maximum result set of 200 exceeded. Consider refining.

Search within Results

> Advanced Search

> Search Help/Tips

Sort by: Title Publication Publication Date Score

Results 1 - 20 of 200 short listing

Prev Page 1 2 3 4 5 6 7 8 9 10 Next Page

1 A compression technique to materialize transitive closure 100%
H. V. Jagadish
ACM Transactions on Database Systems (TODS) December 1990
Volume 15 Issue 4
An important feature of database support for expert systems is the ability of the database to answer queries regarding the existence of a path from one node to another in the directed graph underlying some database relation. Given just the database relation, answering such a query is time-consuming, but given the transitive closure of the database relation a table look-up suffices. We present an indexing scheme that permits the storage of the pre-computed transitive closure of a database re ...

2 Modeling the cost of resource allocation in distributed control 100%
Martin D. Fraser , Ross A. Gagliano , Mark E. Schaefer
ACM SIGSIM Simulation Digest , Proceedings of the 23rd annual symposium on Simulation April 1990
Volume 20 Issue 4
Modifying our previously developed simulation model [FRA89], we study in this paper the costs associated with distributed allocation of computing resources in a multitasking environment. Using funds endowed upon arrival, computing tasks compete for necessary resources through sealed-bid auctions to improve their processing schedules. The costs and times dedicated to auctioning are compared to the costs and times allowed for task processing.

Measuring computing resources in terms of processi ...

3 Parameter-passing and the lambda calculus 100%
Erik Crank , Matthias Felleisen
Proceedings of the 18th ACM SIGPLAN-SIGACT symposium on Principles of programming languages January 1991

4 Pointer-induced aliasing: a problem taxonomy 100%
William Landi , Barbara G. Ryder
Proceedings of the 18th ACM SIGPLAN-SIGACT symposium on Principles of programming languages January 1991

5 Program optimization and parallelization using idioms 100%
Shlomit S. Pinter , Ron Y. Pinter
Proceedings of the 18th ACM SIGPLAN-SIGACT symposium on Principles of programming languages January 1991

6 Automatic construction of sparse data flow evaluation graphs 100%
Jong-Deok Choi , Ron Cytron , Jeanne Ferrante
Proceedings of the 18th ACM SIGPLAN-SIGACT symposium on Principles of programming languages January 1991

7 A theoretical analysis of feedback flow control 100%
S. Shenker
ACM SIGCOMM Computer Communication Review , Proceedings of the ACM symposium on Communications architectures & protocols August 1990
Volume 20 Issue 4
Congestion is a longstanding problem in datagram networks. One congestion avoidance technique is feedback flow control, in which sources adjust their transmission rate in response to congestion signals sent (implicitly or explicitly) by network gateways. The goal is to design flow control algorithms which provide time-scale invariant, fair, stable, and robust performance. In this paper we introduce a simple model of feedback flow control, in which sources make synchronous rate adjus ...

8 Parametric performance evaluation of a micro-based database system 100%
Ray Hashemi-Nassab
Proceedings of the 1990 ACM SIGSMALL/PC symposium on Small systems February 1990
An extended analytical technique for performance evaluation of software packages is introduced. The purpose of this analytical technique is to present the performance evaluation of a system in parametric equations. The equations can be easily used in studying the system behavior in different environments without re-evaluating the performance of the system. This technique was applied on a micro-based relational database system called MICREL. The performance criteria were the transaction dela ...

9 Replacing function parameters by global variables 100%
Peter Sestoft

Proceedings of the fourth international conference on Functional programming languages and computer architecture November 1990

10 Unidraw: a framework for building domain-specific graphical editors 100%

John M. Vlissides , Mark A. Linton

ACM Transactions on Information Systems (TOIS) July 1990

Volume 8 Issue 3

Unidraw is a framework for creating graphical editors in domains such as technical and artistic drawing, music composition, and circuit design. The Unidraw architecture simplifies the construction of these editors by proving programming abstractions that are common across domains. Unidraw defines four basic abstractions: components define operations on components, and external representations define the mapping between components and the file format generat ...

11 A tour of suite user interface software 100%

Prasun Dewan

Proceedings of the 3rd annual ACM SIGGRAPH symposium on User interface software and technology August 1990

12 The rendering architecture of the DN10000VS 100%

David Kirk , Douglas Voorhies

ACM SIGGRAPH Computer Graphics , Proceedings of the 17th annual conference on Computer graphics and interactive techniques September 1990

Volume 24 Issue 4

13 Managing a diamond jewelry manufacturing business using APL 100%

Steven I. Promisel , James V. Merrill

ACM SIGAPL APL Quote Quad , Conference proceedings on APL 90: for the future May 1990

Volume 20 Issue 4

We discuss the development of several interconnected computer systems designed and built for a manufacturer of finished jewelry. Our approach will be two-fold. We describe how the company uses APL-based computer systems to analyze pricing issues and handle the manufacturing process. We show how the system has been developed by describing what techniques have been employed and what tools have been created and used. The purpose of this paper is to show how an APL application development syste ...

14 An application development platform 100%

Morten Kromberg , Martin Gfeller

ACM SIGAPL APL Quote Quad , Conference proceedings on APL 90: for the future May 1990

Volume 20 Issue 4

While APL was a highly productive system for personal computing and application development when it was introduced, it is lacking nowadays in the areas of programming-in-the-large and user interface construction. The design and implementation of a new Application Development Platform (ADP) is described. The ADP frees the programmer from user interface concerns, and frees the application from device specific code. The ADP ensures consistent user interfaces based on state-of-the-ar ...

15 Emerging technologies: WLANs and WPANs: On flow reservation and admission control for distributed scheduling strategies in IEEE802.11 wireless LAN 100%
Ming Li , B. Prabhakaran , Sathish Sathyamurthy
Proceedings of the 8th international workshop on Modeling analysis and simulation of wireless and mobile systems September 2003
Providing service differentiation in IEEE802.11 Wireless LANs [3] has been investigated by many researchers ([2], [5], [6], [7], [9], [13]). It has been shown [1] that some distributed schedulers such as DFS [9] and EDCF [5] can achieve high throughput and certain service differentiation comparing to DCF and PCF provided that the traffic load in the system is low or medium. However, those strategies do not support flow reservation and thus cannot guarantee QoS requirements of high priority real- ...

16 Ad hoc and sensor networks: Flooding strategy for target discovery in wireless networks 100%
Zhao Cheng , Wendi B. Heinzelman
Proceedings of the 8th international workshop on Modeling analysis and simulation of wireless and mobile systems September 2003

17 Consistancy and replication: TiNA: a scheme for temporal coherency-aware in-network aggregation 100%
Mohamed A. Sharaf , Jonathan Beaver , Alexandros Labrinidis , Panos K. Chrysanthis
Proceedings of the 3rd ACM international workshop on Data engineering for wireless and mobile access September 2003
This paper presents TiNA, a scheme for minimizing energy consumption in sensor networks by exploiting end-user tolerance to temporal coherency. TiNA utilizes temporal coherency tolerances to both reduce the amount of information transmitted by individual nodes (communication cost dominates power usage in sensor networks), and to improve quality of data when not all sensor readings can be propagated up the network within a given time constraint. TiNA was evaluated against a traditional in-network ...

18 Automated test generation for industrial Erlang applications 100%
Johan Blom , Bengt Jonsson
Proceedings of the 2003 ACM SIGPLAN workshop on Erlang August 2003
We present an implemented technique for generating test cases from state machine specifications. The work is motivated by a need for testing of protocols and services developed by the company Mobile Arts. We have developed a syntax for description of state machines extended with data variables. From such state machines, test cases are generated by symbolic execution. The test cases are symbolically represented; concrete test cases are generated by instantiation of data parameters.

19 Context-sensitive slicing of concurrent programs 100%
Jens Krinke
ACM SIGSOFT Software Engineering Notes , Proceedings of the 9th European software engineering conference held jointly with 10th ACM SIGSOFT international symposium on Foundations of software engineering September 2003
Volume 28 Issue 5
Program slicing is a technique to identify statements that may influence the computations at other statements. Precise slicing has been shown to be undecidable for concurrent programs.

This work presents the first context-sensitive approach to slice concurrent programs accurately. It extends the well known structures of the control flow graph and the (interprocedural) program dependence graph for concurrent programs with interference. This new technique does not require serialization or inlining ...

20 A strategy for efficiently verifying requirements 100%

Ralph D. Jeffords , Constance L. Heitmeyer

ACM SIGSOFT Software Engineering Notes , Proceedings of the 9th European software engineering conference held jointly with 10th ACM SIGSOFT international symposium on Foundations of software engineering September 2003

Volume 28 Issue 5

This paper describes a compositional proof strategy for verifying properties of requirements specifications. The proof strategy, which may be applied using either a model checker or a theorem prover, uses known state invariants to prove state and transition invariants. Two proof rules are presented: a standard incremental proof rule analogous to Manna and Pnueli's incremental proof rule and a compositional proof rule. The advantage of applying the compositional rule is that it decomposes a large ...

Search Results

Search Results for: **[(vhdl or verliog or hdl or (hardware <sentence> language))<AND>((((module* or sub*routine* or header* or function* or procedure*) <paragraph> ((initial* or assign* or set* or defin* or pre*determin*) <paragraph> ((initial or starting or first or beginning) <sentence> (node* or signal* or variable* or wire* or port* or reg* or register*) <sentence> (condition* or value* or voltage* or state*))))))]**
Found **247** of **123,929 searched.**

Warning: Maximum result set of 200 exceeded. Consider refining.

Search within Results

> Advanced Search

> Search Help/Tips

Sort by: Title Publication Publication Date Score

Results 1 - 20 of 200 short listing

Prev Page 1 2 3 4 5 6 7 8 9 10 Next Page

1 Program optimization and parallelization using idioms 100%
Shlomit S. Pinter , Ron Y. Pinter
Proceedings of the 18th ACM SIGPLAN-SIGACT symposium on Principles of programming languages January 1991

2 Design of the Mneme persistent object store 100%
J. Eliot B. Moss
ACM Transactions on Information Systems (TOIS) April 1990
Volume 8 Issue 2
The Mneme project is an investigation of techniques for integrating programming language and database features to provide better support for cooperative, information-intensive tasks such as computer-aided software engineering. The project strategy is to implement efficient, distributed, persistent programming languages. We report here on the Mneme persistent object store, a fundamental component of the project, discussing its design and initial prototype. Mneme stores objects

3 Verification, analysis of embedded systems: Verification of design decisions in ForSyDe 100%
Tarvo Raudvere , Ingo Sander , Ashish Kumar Singh , Axel Jantsch
Proceedings of the 1st IEEE/ACM/IFIP international conference on Hardware/software codesign & system synthesis October 2003

The ForSyDe methodology has been developed for system level design. Starting with a formal specification model that captures the functionality of the system at a high abstraction level, it provides formal design transformation methods for a transparent refinement process of the specification model into an implementation model that is optimized for synthesis. A transformation may be semantic preserving or a design decision. The latter modifies the semantics of the system level description and cha ...

4 Onward papers: Routine run-time code generation 100%

Sam Kamin

Companion of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications October 2003

Run-time code generation (RTCG) would be used routinely if application programmers had a facility with which they could easily create their own run-time code generators, because it would offer benefits both in terms of the efficiency of the code that programmers would produce and the ease of producing it. Such a facility would necessarily have the following properties: it would not require that programmers know assembly language; programmers would have full control over the generated code; the c ...

5 Puzzling with microcode 100%

Jeremy Jones

ACM SIGARCH Computer Architecture News December 1983

Volume 11 Issue 5

A Pascal version of the puzzle program was executed on a user-microprogrammable HLL machine and a "runtime profile" obtained. The code sections where most of the execution time was spent were found and replaced with microcode. A six fold increase in execution speed was achieved by writing 25 microinstructions.

6 The Honeywell Modular Microprogram Machine: M3 100%

E. Douglas Jensen , Richard Y. Kain

ACM SIGARCH Computer Architecture News , Proceedings of the 4th annual symposium on Computer architecture March 1977

Volume 5 Issue 7

M3 is intended for research into unconventional special purpose stored program elements of computer systems (for example, a distributed computer Bus Interface Unit). The principal requirements for such a machine are flexibility and modularity. M3 consists of an application independent Kernel Machine to which application-dependent Functional Modules are attached. The Kernel Machine is vertically microprogrammed; it includes highly capable microinstru ...

7 Digital control simulation system 100%

H. Rex Hartson

Proceedings of the 6th annual conference on Design Automation January 1969

Today there is widespread application of digital control circuitry in a wide range of products. This paper describes a simulation system in which the designer of these control circuits can interact with his design ideas before they are implemented in hardware. The Digitial Control Simulation System (DCSS) is a digital design description language with a set of programs to generate and execute a simulation program. The main use of this system (with an appropriate hardware interface ...

8 Now you may compose temporal logic specifications 100%

Howard Barringer , Ruurd Kuiper , Amir Pnueli

Proceedings of the sixteenth annual ACM symposium on Theory of computing December 1984

A compositional temporal logic proof system for the specification and verification of concurrent programs is presented. Versions of the system are developed for shared variables and communication based programming languages that include procedures.

9 A general-purpose high-level language machine for minicomputers 100%

Bradford W. Wade , Victor B. Schneider

ACM SIGPLAN Notices , Proceedings of the meeting on SIGPLAN/SIGMICRO interface May 1973

Volume 9 Issue 8

In the course of our investigations into the design of translator writing systems (compiler-compilers), it has been established [2] that a certain set of “semantic primitives” can adequately express the major portion of the semantics of programs written in any of the several common high-level languages (e.g., PL/I, ALGOL W). It was also observed that each of these semantic primitives, while representing frequently-used high-level language constructs, corresponds to predictable s ...

10 The design of an emulator for a parallel machine language 100%

Victor R. Lesser

ACM SIGPLAN Notices , Proceedings of the meeting on SIGPLAN/SIGMICRO interface May 1973

Volume 9 Issue 8

A paradigm is developed for structuring a complex emulator operating in a parallel hardware environment. This paradigm is based on the view that a complex emulator is best structured as of a set of microprocesses, each performing a small independent task, that interact in a closely-coupled manner. This is in contrast to the conventional method of structuring an emulator as a set of subroutines with a sequential flow of control among them. The design of an emulator for a parallel machine lan ...

11 Technical correspondence: Assessment of the Java programming language for use in high integrity systems 100%

Jagun Kwon , Andy Wellings , Steve King

ACM SIGPLAN Notices April 2003

Volume 38 Issue 4

This paper sets a goal of investigating the use of Java in the development of high integrity systems. Based on previous studies, guidelines, and standards, we develop 23 criteria that are used for the following assessment of Java. A summary of the assessment is provided before we go on to review a few existing subsets of the language.

12 Distributed communication via global buffer 100%

David Gelernter , Arthur J. Bernstein

Proceedings of the first ACM SIGACT-SIGOPS symposium on Principles of distributed computing August 1982

Design and implementation of an inter-address-space communication mechanism for the SBN

network computer are described. SBN's basic communication primitives appear in context of a new distributed systems programming language strongly supported by the network communication kernel. A model in which all communication takes place via a distributed global buffer results in simplicity, generality and power in the communication primitives. Implementation issues raised by the requirements of the gl ...

13 Digital Simulation of an Aerospace Vehicle 100%

J. R. Mitchell , J. W. Moore , H. H. Trauboth

Proceedings of the 1967 22nd national conference January 1967

The rapid development of computer technology and the creation of new engineering oriented languages have established that general purpose digital computers are increasingly suitable for simulation of the dynamics of large physical systems. In the Aerospace Vehicle Simulation (AVS) Program, an effort has been undertaken at Marshall Space Flight Center (MSFC), Huntsville, Alabama, to simulate continuous and discrete dynamics of an aerospace vehicle and its ground support equipment (GSE) on a ...

14 The specification of program flow in Madcap 6 100%

James B. Morris , Mark B. Wells

Proceedings of the ACM annual conference - Volume 2 August 1972

The control structures of the Madcap language have evolved to a point where today those of Madcap 6 have obviated programmer defined labels and go-to statements. The benefits of the removal of these concepts are discussed in detail. Madcap has a powerful class of data structures, including sets, sequences, and expressions, along with a full array of operators for manipulating these structures. These operators include important facilities for forming sets and for forming and concatenating se ...

15 Assignments and high level data types 100%

Bengt Nordström

Proceedings of the 1978 annual conference - Volume 2 January 1978

If x and y are integer variables or any other variables of a “simple” type, then the meaning of the assignment x:&equil;y is quite similar in different programming languages. The value of y is copied (into a register) and then stored into x so that a fresh copy of the value of y becomes the current value of x. On the other hand if x and y are compound objects the the situation is quite different: some languages don't allow assignments at all, and other languages call for a condi ...

16 Digital logic modeling system based on MODLAN 100%

Adam Pawlak

Proceedings of the nineteenth design automation conference January 1982

Digital logic modeling system based on MODLAN - the language for multilevel description and simulation has been presented in this paper. The basic language notions and concepts of structural, functional and behavioral descriptions are recapitulated. MODLAN short presentation has been illustrated with examples of digital elements description Two problems of the system implementation have been dealt with; one is that of data ba ...

17 An Interactive Simulation System for structured logic design—ISS 100%

Takeshi Sakai , Yoshiyuki Tsuchida , Hiroto Yasuura , Yasushi Ooi , Yoshitsugu Ono , Hiroshi Kano , Shinji Kimura , Shuzo Yajima

Proceedings of the nineteenth design automation conference January 1982
An Interactive Simulation System (ISS) is presented. ISS is an integrated interactive CAD system for logic design, and is configurated “module oriented” to support structured logic design. An Interactive Simulator (IS) is used for design verification. A designer can control simulation steps interactively in IS, and he can find design errors early using a good interactive interface. A Structured Hardware Design Language (SHDL) is used to describe logic designs.

18 The conlan project: Status and future plans 100%
Robert Piloty , Dominique Borrione
Proceedings of the nineteenth design automation conference January 1982
CONLAN (CONsensus LANguage) is a general formal language construction mechanism for the description of hard- and firmware at different levels of abstraction. It has been developed by the international CONLAN Working Group. Members of the CONLAN language family are derived from a common root language called BCL (Base ConLan). This language provides the basic object types and operations to describe the behavior and the structure of digital systems in space and time. The paper is based on the ...

19 Interactive design language: A unified approach to hardware simulation, synthesis and documentation 100%
L. I. Maissel , D. L. Ostapko
Proceedings of the nineteenth design automation conference January 1982
IDL is a hardware design language in use in the VLSI environment. It incorporates a significant number of high-level features such as groups, subroutines, and labels and is particularly well adapted to dealing with parallelism at the hardware level. In addition to being human intelligible (and therefore appropriate as a documentation medium), IDL code can be used to generate 2-level logic which, under the IDL system, can be manipulated in a number of ways, including product term factoring a ...

20 Designing with LCD: language for computer design 100%
C. J. Evangelisti , G. Goertzel , H. Ofek
Proceedings of the 14th design automation conference January 1977
LCD, a Language for Computer Design, permits the description of both the function and the structure of digital hardware. Machines may be described at a very high (system) level, at a very low (gate) level, or at any intermediate level. LCD descriptions are processed by a set of programs which support high level design and verification. These programs include facilities to check the description for correct syntax, carry out a mixed numeric and symbolic simulation, check a dataflow ...

Search Results

Search Results for: **[((releas* or free*) <paragraph> ((initial or starting or first or beginning) <sentence> (node* or signal* or variable* or wire* or port* or reg* or register*) <sentence> (condition* or value* or voltage* or state*)))<AND>(((vhdl or verliog or hdl or (hardware <sentence> language))<AND>((((module* or sub*routine* or header* or function* or procedure*) <paragraph> ((initial* or assign* or set* or defin* or pre*determin*) <paragraph> ((initial or starting or first or beginning) <sentence> (node* or signal* or variable* or wire* or port* or reg* or register*) <sentence> (condition* or value* or voltage* or state*))))))))]**
Found **23** of **123,929 searched.**

Search within Results

Sort by: Title Publication Publication Date Score Binder

Results 1 - 20 of 23 short listing

Prev Page 1 2 Next Page

1 Digital control simulation system 100%
H. Rex Hartson
Proceedings of the 6th annual conference on Design Automation January 1969
Today there is widespread application of digital control circuitry in a wide range of products. This paper describes a simulation system in which the designer of these control circuits can interact with his design ideas before they are implemented in hardware. The Digitial Control Simulation System (DCSS) is a digital design description language with a set of programs to generate and execute a simulation program. The main use of this system (with an appropriate hardware interface ...

2 A general-purpose high-level language machine for minicomputers 100%
Bradford W. Wade , Victor B. Schneider
ACM SIGPLAN Notices , Proceedings of the meeting on SIGPLAN/SIGMICRO interface May 1973
Volume 9 Issue 8
In the course of our investigations into the design of translator writing systems (compiler-compilers), it has been established [2] that a certain set of “semantic primitives” can adequately express the major portion of the semantics of programs written in any of the several common high-level languages (e.g., PL/I, ALGOL W). It was also observed that each of these semantic primitives, while representing frequently-used high-level language constructs, corresponds to predictable s ...

3 Digital Simulation of an Aerospace Vehicle 100%

J. R. Mitchell , J. W. Moore , H. H. Trauboth

Proceedings of the 1967 22nd national conference January 1967

The rapid development of computer technology and the creation of new engineering oriented languages have established that general purpose digital computers are increasingly suitable for simulation of the dynamics of large physical systems. In the Aerospace Vehicle Simulation (AVS) Program, an effort has been undertaken at Marshall Space Flight Center (MSFC), Huntsville, Alabama, to simulate continuous and discrete dynamics of an aerospace vehicle and its ground support equipment (GSE) on a ...

4 The SLIDE simulator: A facility for the design and analysis of computer interconnections 100%

Arthur H. Altman , Alice C. Parker

Proceedings of the seventeenth design automation conference on Design automation June 1980

Interconnection design can have a profound effect on the price and performance of a digital system. This paper describes a new simulation facility that is designed to allow the user to describe and simulate the behavior of an interconnected system. The simulator provides the capability to devise, debug, and evaluate digital interconnection schemes. The user first writes a description of the system interconnections using the hardware descriptive language SLIDE. The UNIBUS, for exa ...

5 Four models for the analysis and optimization of program control structures 100%

Terrence Pratt

Proceedings of seventh annual ACM symposium on Theory of computing May 1975

The analysis of the relation between the structure of a program and the function that it computes requires a decomposition of the program into its components. Traditionally this decomposition has been based on the common division of a program into subprograms, and ultimately into statements, expressions and individual variables and constants. In this paper an alternative decomposition is proposed that is based on the decomposition of a program into a set of kernel elements

6 Some implications of shared variables 100%

Adin D. Falkoff

Proceedings of the eighth international conference on APL September 1976

Recognition of shared variables as the fundamental means of communication between concurrently operating processes has important implications for the design of systems and programming languages. As a consequence, current APL systems, such as APLSV, embody two shared variable facilities: one for establishing temporary interfaces between system components, and the other one for defining parts of the permanent interface between an APL program and the underlying processor. These are treated in ...

7 VVDS: a verification/diagnosis system for VHDL 100%

H. T. Liaw , K.-T. Tran , C.-S. Lin

Proceedings of the 1989 26th ACM/IEEE conference on Design automation conference June 1989

In this paper, an interactive verification and diagnosis system for VHDL [Vm88], VVDS, is presented. In VVDS, hybrid simulation, which simulates with both numerical and symbolic

data, is implemented to achieve an effective compromise of the enormous quantity of input test data in the conventional simulation and the complexity of symbolic expression in the symbolic execution. To support efficient user interface in the verification and diagnosis process, both on-line programming of commands a ...

8 A compiled implementation of strong reduction 100%

Benjamin Grégoire , Xavier Leroy

ACM SIGPLAN Notices , Proceedings of the seventh ACM SIGPLAN international conference on Functional programming September 2002

Volume 37 Issue 9

Motivated by applications to proof assistants based on dependent types, we develop and prove correct a strong reducer and ß-equivalence checker for the ?-calculus with products, sums, and guarded fixpoints. Our approach is based on compilation to the bytecode of an abstract machine performing weak reductions on non-closed terms, derived with minimal modifications from the ZAM machine used in the Objective Caml bytecode interpreter, and complemented by a recursive "read back" procedure. ...

9 Compilers and Optimization: A new method for compiling schizophrenic synchronous programs 100%

K. Schneider , M. Wenz

Proceedings of the international conference on Compilers, architecture, and synthesis for embedded systems November 2001

Synchronous programming languages have proved to be advantageous for designing software and hardware for embedded systems. Despite their clear semantics, their compilation is remarkably difficult: In particular, one has to take care of potential schizophrenia problems. Although these problems are correctly translated with existing compilers, there is still a need for clean algorithms. In this paper, we present the first solution to eliminate schizophrenia problems by program transformations. The ...

10 Systems semantics: principles, applications, and implementation 100%

Ray Boute

ACM Transactions on Programming Languages and Systems (TOPLAS) January 1988

Volume 10 Issue 1

Systems semantics extends the denotational semantics of programming languages to a semantics for the description of arbitrary systems, including objects that are not computations in any sense. By defining different meaning functions, the same formal description may be used to denote different system properties, such as structure, behavior, component cost, and performance aspects (e.g., timing). The definition of these semantic functions also provides guidance in language design, ...

11 Design of a Machine-Independent Optimizing System for Emulator Development 100%

Perng-Ti Ma , T. G. Lewis

ACM Transactions on Programming Languages and Systems (TOPLAS) April 1980

Volume 2 Issue 2

Methods are described to translate a certain machine-independent intermediate language (IML) to efficient microprograms for a class of horizontal microprogrammable machines. The IML is compiled directly from a high-level microprogramming language used to implement a virtual instruction set processor as a microprogram. The primary objective of the IML-to-host

machine interface design is to facilitate language portability. Transportability is accomplished by use of a field descript ...

12 Data speculation support for a chip multiprocessor 100%
Lance Hammond , Mark Willey , Kunle Olukotun
Proceedings of the eighth international conference on Architectural support for programming languages and operating systems October 1998
Volume 32 , 33 Issue 5 , 11
Thread-level speculation is a technique that enables parallel execution of sequential applications on a multiprocessor. This paper describes the complete implementation of the support for threadlevel speculation on the Hydra chip multiprocessor (CMP). The support consists of a number of software speculation control handlers and modifications to the shared secondary cache memory system of the CMP This support is evaluated using five representative integer applications. Our results show that the s ...

13 Isomorph-free model enumeration: a new method for checking relational specifications 100%
Daniel Jackson , Somesh Jha , Craig A. Damon
ACM Transactions on Programming Languages and Systems (TOPLAS) March 1998
Volume 20 Issue 2
Software specifications often involve data structures with huge numbers of value, and consequently they cannot be checked using standard state exploration or model-checking techniques. Data structures can be expressed with binary relations, and operations over such structures can be expressed as formulae involving relational variables. Checking properties such as preservation of an invariant thus reduces to determining the validity of a formula or, equivalently, finding a model (of the form ...

14 Object graph rewriting: an experimental parallel implementation 100%
John Glauert
Proceedings of the second international symposium on Parallel symbolic computation July 1997

15 Interprocedural register allocation for lazy functional languages 100%
Urban Boquist
Proceedings of the seventh international conference on Functional programming languages and computer architecture October 1995

16 Performance specification using attributed grammars 100%
Ram Mandayam , Ranga Vemuri
Proceedings of the 30th international on Design automation conference July 1993

17 Flexible timing specification in a VHDL synthesis subset 100%
A. Stoll , J. Biesenack , S. Rumler
Proceedings of the conference on European design automation November 1992

18 A technique to generate feasible tests for communications systems with multiple timers 89%
Mariusz A. Fecko , M. Ümit Uyar , Ali Y. Duale , Paul D. Amer
IEEE/ACM Transactions on Networking (TON) October 2003

Volume 11 Issue 5
We present a new model for testing real-time protocols with multiple timers, which captures complex timing dependencies by using simple linear expressions involving timer-related variables. This new modeling technique, combined with the algorithms to eliminate inconsistencies, allows generation of feasible test sequences without compromising their fault coverage. The model is specifically designed for testing to avoid performing full reachability analysis, and to control the growth of the number ...

19 Dynamic scheduling and synchronization synthesis of concurrent digital systems under system-level constraints 87%
Claudionor N. Coelho , Giovanni De Micheli
Proceedings of the 1994 IEEE/ACM international conference on Computer-aided design
November 1994
We present in this paper a novel control synthesis technique for system-level specifications that are better described as a set of concurrent synchronous descriptions, their synchronizations and constraints. The proposed synthesis technique considers the degrees of freedom introduced by the concurrent models and by the environment in order to satisfy the design constraints.Synthesis is divided in two phases. In the first phase, the original specification is translated into an alg ...

20 Behavioral synthesis: Coordinated transformations for high-level synthesis of high performance microprocessor blocks 85%
Sumit Gupta , Nick Savoiu , Nikil Dutt , Rajesh Gupta , Alex Nicolau , Timothy Kam , Michael Kishinevsky , Shai Rotem
Proceedings of the 39th conference on Design automation June 2002
High performance microprocessor designs are partially characterized by functional blocks consisting of a large number of operations that are packed into very few cycles (often single-cycle) with little or no resource constraints but tight bounds on the cycle time. Extreme parallelization, conditional and speculative execution of operations is essential to meet the processor performance goals. However, this is a tedious task for which classical high-level synthesis (HLS) formulations are inadequa ...

21 Declaration of unknowns in DAE-based hybrid system specification 80%

D. A. van Beek , V. Bos , J. E. Rooda

ACM Transactions on Modeling and Computer Simulation (TOMACS) January 2003 Volume 13 Issue 1

The majority of hybrid languages are based on the assumption that discontinuities in differential variables at discrete events are modeled by explicit mappings. When there are algebraic equations restricting the allowed new values of the differential variables, explicit remapping of differential variables forces the modeler to solve the algebraic equations. To overcome this difficulty, hybrid languages use many different language elements. This article shows that only one language element is nee ...

22 Automatic synthesis of extended burst-mode circuits using generalized C-elements 80%

K. Yun

Proceedings of the conference on European design automation September 1996

23 Session 6C: Markovian analysis and asynchronous circuits: Achieving fast and exact hazard-free logic minimization of extended burst-mode gC finite state machines 77%

Hans Jacobson , Chris Myers , Ganesh Gopalakrishnan

Proceedings of the 2000 IEEE/ACM international conference on Computer-aided design November 2000

This paper presents a new approach to two-level hazard-free logic minimization in the context of extended burst-mode finite state machine synthesis targeting generalized C-elements (gC). No currently available minimizers for literal-exact two-level hazard-free logic minimization of extended burst-mode gC controllers can handle large circuits without synthesis times ranging up over thousands of seconds. Even existing heuristic approaches take too much time when iterative exploration over a large ...

Your search matched **3** of **764280** documents.
Results are shown **25** to a page, sorted by **publication year** in **descending** order.
You may refine your search by editing the current search expression or entering a new one the te

Then click **Search Again**.

VHDL and signal and monitor

Search Again

Results:
Journal or Magazine = **JNL** Conference = **CNF** Standard = **STD**

1 **An integrated trigger system for the PIERRE AUGER fluorescence de**
Gemmeke, H.; Klinger, V.; Tchemiakhovski, D.
Nuclear Science Symposium, 1997. IEEE , 1997
Page(s): 71 -75 vol.1

[Abstract] [PDF Full-Text (424 KB)] **CNF**

2 **Mixed-level modeling in VHDL using the watch-and-react interface**
Dungan, W.W.; Klenke, R.H.; Aylor, J.H.
VHDL International Users' Forum, 1997. Proceedings , 1997
Page(s): 25 -32

[Abstract] [PDF Full-Text (692 KB)] **CNF**

3 **Monitoring human powered submarine performance using a data ac system with a custom display ASIC**
Guy, B.M., III.; Srinivasan, A.K.
Southeastcon '95. Visualize the Future., Proceedings., IEEE , 1995
Page(s): 413 -416

[Abstract] [PDF Full-Text (384 KB)] **CNF**

WEST

Edit Saved Searches for User *asharon*

Queries 261 through 310.

Next | Oldest

Update | Cancel | Help | Main Menu | Logout

Delete: | Delete ALL

S #	Comment	Database	Query String	Dele
S310		USPT	((((((716/4.CCLS.) or (716/5.CCLS.) or (716/6.CCLS.) or (716/18.ccls.))and (user))and (clock adj cycle$2))and waveform)and (user adj (defined or determined or selected or chosen))) and (number adj3 clock adj cycle adj3 (waveform or plot or graph))	
S309		USPT	(((((716/4.CCLS.) or (716/5.CCLS.) or (716/6.CCLS.) or (716/18.ccls.))and (user))and (clock adj cycle$2))and waveform) and (user adj (defined or determined or selected or chosen))	
			((((716/4.CCLS.) or (716/5.CCLS.)	

Your search matched **2** of **764280** documents.
Results are shown **25** to a page, sorted by **publication year** in **descending** order.
You may refine your search by editing the current search expression or entering a new one the te
Then click **Search Again**.

(Verilog or HDL) and signal and monitor

Search Again

Results:
Journal or Magazine = **JNL** Conference = **CNF** Standard = **STD**

1 **A high level language implementation of a general purpose telemetr for biomedical applications**
Rorie, J.F.; Simmerman, J.K.; Makki, R.Z.; Piendl, R.D.
System Theory, 1996., Proceedings of the Twenty-Eighth Southeastern Symp 1996
Page(s): 338 -342

[Abstract] [PDF Full-Text (564 KB)] **CNF**

2 **Postlayout simulation of FPGAs using verilog**
Colley, M.J.
Software Support and CAD Techniques for FPGAs, IEE Colloquium on , 1994
Page(s): 6/1 -6/3

[Abstract] [PDF Full-Text (152 KB)] **CNF**

Your search matched **28** of **764280** documents.
Results are shown **15** to a page, sorted by **publication year** in **descending** order.
You may refine your search by editing the current search expression or entering a new one the te
Then click **Search Again**.

((VHDL)or (Verilog)) and(waveform)

Search Again

Results:
Journal or Magazine = **JNL** Conference = **CNF** Standard = **STD**

1 **FPGA implementation of PWM pattern generators**
Dan Deng; Su Chen; Joos, G.
Electrical and Computer Engineering, 2001. Canadian Conference on , Volume

Page(s): 1279 -1284 vol.2

[Abstract] [PDF Full-Text (1312 KB)] **CNF**

2 **FPGA implementation of PWM pattern generators [for PWM invertor**
Dan Deng; Su Chen; Joos, G.
Electrical and Computer Engineering, 2001. Canadian Conference on , Volume

Page(s): 225 -230 vol.1

[Abstract] [PDF Full-Text (1276 KB)] **CNF**

3 **Substrate noise generation in complex digital systems: efficient mod and simulation methodology and experimental verification**
Van Heijningen, M.; Badaroglu, M.; Donnay, S.; De Man, H.; Gielen, G.; Enge Bolsens, I.
Solid-State Circuits Conference, 2001. Digest of Technical Papers. ISSCC. 200 International , 2001
Page(s): 342 -343, 463

[Abstract] [PDF Full-Text (216 KB)] **CNF**

4 **Direct digital frequency synthesizer architecture based on Chebyshe approximation**